

Code:

```
using UnityEngine;
using System.Collections;

[RequireComponent(typeof(ComboInventory))]
public class ComboManager : MonoBehaviour {
    //Animation
    Animator anim;

    //using classes
    private ComboDefinition comboDefinition;
    private ComboInventory comboInventory;

    // declaring variables
    private int comboCount = 0;
    private int availableCombos;
    private int currentlyAvailableCombos;
    private int comboListLength;
    private int componentIndex;
    private float timeLastButtonPressed;
    private float timeAllowed;
    // private bool checkTime;
    //private bool proceed;
    // private int colorCountDown;

    void Awake() {
        comboInventory = GetComponent<ComboInventory>();
        anim = GetComponent<Animator>();
    }
    void Start() {
        //initializing variables
        //colorCountDown = 120;
        //proceed = true;
        timeAllowed = 0.5f;
        // checkTime = false;
        componentIndex = -1;
        comboListLength = comboInventory.comboList.Length;
        availableCombos = comboInventory.comboList.Length; //this has to change
        currentlyAvailableCombos = availableCombos; // keeps track of what is

    }
    void Update() {
        if (Input.anyKeyDown) {
            string button = ComboInputManager.SaveInput();
            CheckTime();
            CheckForCombos(button);
        }
    }

    void CheckForCombos(string b) {

        componentIndex++;

        // check each combo
        for (int i = 0; i < comboListLength; i++) {
            int comboLength = comboInventory.comboList[i].components.Length;
            //check if combos are failed or available
        }
    }
}
```

```

        if ((!comboInventory.comboList[i].failed) && (comboInventory.comboList[i].progress < comboInventory.comboList[i].complete))
            // checking for a button match
            if (b == comboInventory.comboList[i].components[componentIndex])
                comboInventory.comboList[i].progress++;

                //checking if combo is complete or in progress
                if (comboInventory.comboList[i].progress >= comboInventory.comboList[i].complete)
                    Debug.Log("combo COMPLETE");
                    comboCount++;

                    //Combo Finisher Ground Eruptions
                    anim.SetTrigger (comboInventory.comboList[i].components[componentIndex].animName);
                    //Combo Finisher Ground Wave
                    anim.SetTrigger (comboInventory.comboList[i].components[componentIndex].animName);
                    //Combo Finisher ChainSaw Blade
                    anim.SetTrigger (comboInventory.comboList[i].components[componentIndex].animName);
                    //Combo Finisher Back Strike
                    anim.SetTrigger (comboInventory.comboList[i].components[componentIndex].animName);

                    ResetComboChecking();
                    break;
            }else {
                //Debug.Log("combo in Progress");
            }//end of if combo is complete or in progress

        }else {
            comboInventory.comboList[i].failed = true;
            //Debug.Log("combo Failed");
            currentlyAvailableCombos--;

            if (currentlyAvailableCombos <= 0) {
                //Debug.Log("NO MORE combos possible");
                ResetComboChecking();
                break;
            }

        }//end of checking for a button match
    }//end of if failed/available check
} // end of for loop checking each combo
} // end of checking for combos function

//reset combo checking function
void ResetComboChecking() {
    //    checkTime = false;
    componentIndex = -1;
    currentlyAvailableCombos = availableCombos;
    for (int i = 0; i < comboListLength; i++) {
        comboInventory.comboList[i].progress = 0;
        comboInventory.comboList[i].complete = false;
        comboInventory.comboList[i].failed = false;
    }
    //Debug.Log("Resetting");
}

void CheckTime () {
    //Debug.Log("Time.deltaTime "+Time.time);
}

```

```
//Debug.Log("timeLastButtonPressed + timeAllowed "+timeLastButtonPressed);  
if(Time.time>timeLastButtonPressed + timeAllowed){  
    ResetComboChecking();  
  
}  
timeLastButtonPressed = Time.time;  
}
```